AMENDMENTS TO THE CLAIMS

1. (Withdrawn) A system for autonomically configuring a cluster of nodes, the system comprising:

a knowledge base of workload descriptions and associated configuration parameters;

a genetic computing processor programmed to produce a selection of configuration parameters for a particular workload based upon a set of existing configuration parameters in said knowledge base; and,

a controller coupled to said knowledge base and the cluster of nodes, said controller comprising programming for monitoring the cluster of nodes and for applying individual ones of said selection of configuration parameters to the cluster of nodes to achieve an improved state of operation.

2. (Withdrawn) The system of claim 1, wherein said knowledge base further comprises performance measures associated with said workload descriptions.

3. (Withdrawn) The system of claim 1, wherein said knowledge base further comprises a listing of acceptable configuration parameters which when applied achieve a level of performance which exceeds pre-defined baseline objectives.

4. (Withdrawn) The system of claim 1, wherein said knowledge base further comprises a listing of unacceptable configuration parameters which when applied fail to achieve a level of performance which exceeds pre-defined baseline objectives.

5. (Withdrawn) The system of claim 1, wherein said knowledge base is configured for coupling to a plurality of controllers which are further coupled to a plurality of corresponding clusters of nodes.

6. (Original) A method for autonomically optimizing a cluster of nodes, the method comprising the steps of:

detecting a node in the cluster which requires re-configuration;

identifying a workload hosted by said node and retrieving a set of configuration parameters associated with said workload;

producing a new generation of configuration parameters based upon said retrieved set using a genetic computing process; and,

reconfiguring said node with selected ones of said new generation of configuration parameters.

7. (Original) The method of claim 6, wherein said detecting step comprises the step of detecting at least one condition selected from the group consisting of a node crash, node idleness, node underperformance, and a change in workload hosted in said node.

8. (Currently Amended) The method of claim 6, wherein said producing step comprises the steps of:

performing a crossover operation for said configuration parameters in said retrieved set; and,

mutating at least one element of said configuration parameters in said retrieved set to produce [[a]] the new generation of configuration parameters.

9. (Original) The method of claim 8, wherein said reconfiguring step comprises the steps of:

randomly selecting a new configuration from among said new generation of configuration parameters;

determining whether said randomly selected new configuration is viable; and,

reconfiguring said node with said randomly selected new configuration only if said new configuration is determined to be viable.

10. (Original) The method of claim 9, further comprising the step of writing said randomly selected new configuration to a knowledge base if said randomly selected new configuration is determined to be viable.

11. (Currently Amended) The method of claim 9, further comprising the steps of:

measuring node performance for said reconfigured node; and,

if said reconfigured node fails to meet baseline objectives for performance for said reconfigured node, selecting [[a]] the new configuration for said node and performing said determining and reconfiguring steps for said selected new configuration.

12. (Original) A machine readable storage having stored thereon a computer program for autonomically optimizing a cluster of nodes, the computer program comprising a routine set of instructions for causing the machine to perform the steps of:

detecting a node in the cluster which requires re-configuration;

identifying a workload hosted by said node and retrieving a set of configuration parameters associated with said workload;

producing a new generation of configuration parameters based upon said retrieved set using a genetic computing process; and,

reconfiguring said node with selected ones of said new generation of configuration parameters.

13. (Original) The machine readable storage of claim 12, wherein said detecting step comprises the step of detecting at least one condition selected from the group consisting of a node crash, node idleness, node underperformance, and a change in workload hosted in said node.

14. (Currently Amended) The machine readable storage of claim 12, wherein said producing step comprises the steps of:

performing a crossover operation for said configuration parameters in said retrieved set; and,

mutating at least one element of said configuration parameters in said retrieved set to produce [[a]] the new generation of configuration parameters.

15. (Original) The machine readable storage of claim 14, wherein said reconfiguring step comprises the steps of:

randomly selecting a new configuration from among said new generation of configuration parameters;

determining whether said randomly selected new configuration is viable; and,

reconfiguring said node with said randomly selected new configuration only if said new configuration is determined to be viable.

16. (Original) The machine readable storage of claim 15, further comprising the step of writing said randomly selected new configuration to a knowledge base if said randomly selected new configuration is determined to be viable.

17. (Currently Amended) The machine readable storage of claim 15, further comprising the steps of:

measuring node performance for said reconfigured node; and,

if said reconfigured node fails to meet baseline objectives for performance for said reconfigured node, selecting [[a]] the new configuration for said node and performing said determining and reconfiguring steps for said selected new configuration.